**1. Documentación de usuario final**

**1.1 Guía de usuario**

**Objetivo**: Este sistema permite enviar y recibir mensajes en un chat simple que muestra colores personalizados para cada usuario.

**Pasos para usar la aplicación:**

1. **Abrir la aplicación en el navegador**:
   * Accede al archivo 016-chat con colores.html o ejecuta el servidor en la dirección configurada (192.168.1.151:5001).
2. **Registrar un usuario**:
   * En el campo de texto con el marcador "Introduce tu usuario", escribe tu nombre de usuario.
3. **Enviar un mensaje**:
   * Escribe el mensaje en el campo "Escribe tu mensaje aquí".
   * Presiona Enter para enviar.
4. **Ver mensajes**:
   * Los mensajes aparecerán en el contenedor central con un fondo de color único, basado en el nombre del usuario.
5. **Actualización automática**:
   * El chat se actualiza automáticamente cada segundo para mostrar los mensajes más recientes.

**Ejemplo visual**:

* Usuario: Juan
* Mensaje: Hola a todos
* Color del mensaje: Color generado automáticamente basado en el nombre.

**1.2 FAQ (Preguntas frecuentes)**

**P: ¿Qué hago si no aparecen los mensajes?**  
R: Asegúrate de que el servidor Flask esté ejecutándose en la dirección configurada (192.168.1.151:5001). Verifica también tu conexión a internet.

**P: ¿Cómo funciona el color único de los mensajes?**  
R: El color se genera basándose en la suma de los valores ASCII de las letras del nombre de usuario.

**P: ¿Puedo enviar mensajes sin nombre de usuario?**  
R: No, es obligatorio introducir un usuario para enviar mensajes.

**1.3 Solución de problemas comunes**

* **No se muestran mensajes nuevos**:
  + Verifica que el servidor Flask esté en ejecución.
  + Actualiza la página del navegador.
* **Errores de conexión**:
  + Comprueba que la dirección IP y el puerto configurados en el servidor (192.168.1.151:5001) estén accesibles.

**2. Documentación técnica para desarrolladores**

**2.1 Introducción al software**

El sistema es una aplicación de chat en tiempo real diseñada para prácticas básicas de Flask y JavaScript. Los mensajes se almacenan en memoria mientras el servidor está en ejecución.

**Características principales**:

* Envío y recepción de mensajes.
* Interfaz dinámica basada en HTML, CSS y JavaScript.
* API sencilla para manejar datos.

**Tecnologías utilizadas**:

* Backend: Flask con soporte para CORS.
* Frontend: HTML, CSS y JavaScript.

**2.2 Arquitectura del sistema**

**Componentes principales**:

1. **Servidor Flask** (007-chat.py):
   * Proporciona endpoints /dame y /toma.
   * Maneja mensajes mediante almacenamiento en una lista global.
2. **Interfaz de usuario** (016-chat con colores.html):
   * Envía mensajes al servidor.
   * Recupera mensajes periódicamente para mostrarlos.

**Interacción**:

plaintext

Copiar código

Usuario <--> Frontend (HTML+JS) <--> API Flask <--> Lista de mensajes

**2.3 Estructura del código**

**Backend (007-chat.py)**:

* **Rutas principales**:
  + /: Prueba de conexión.
  + /dame: Retorna todos los mensajes en formato JSON.
  + /toma: Recibe y almacena mensajes enviados por los usuarios.

**Frontend (016-chat con colores.html)**:

* Elementos:
  + input para usuario y mensaje.
  + section para mostrar mensajes.
* Funcionalidad:
  + fetch para interactuar con las rutas del servidor.
  + Actualización automática mediante un temporizador.

**2.4 Especificaciones de API**

* **GET /dame**
  + **Descripción**: Devuelve todos los mensajes en formato JSON.
  + **Respuesta**:

json

Copiar código

[

{"mensaje": "Hola", "usuario": "Juan"},

{"mensaje": "¿Cómo están?", "usuario": "Ana"}

]

* **GET /toma**
  + **Descripción**: Recibe un mensaje y un usuario.
  + **Parámetros**:
    - mensaje (string): El texto del mensaje.
    - usuario (string): El nombre del usuario.
  + **Respuesta**:

json

Copiar código

{"mensaje": "ok"}

**2.5 Dependencias y configuraciones**

* **Dependencias**:
  + Flask
  + Flask-CORS
* **Configuraciones**:
  + Dirección del servidor: 192.168.1.151
  + Puerto: 5001

**3. Guías de instalación y despliegue**

**3.1 Requisitos previos**

* Python 3.6+
* Entorno Flask configurado.
* Navegador web compatible con JavaScript.

**3.2 Instrucciones de instalación**

1. **Instalar dependencias**:

bash

Copiar código

pip install flask flask-cors

1. **Ejecutar el servidor**:

bash

Copiar código

python 007-chat.py

1. **Abrir la interfaz**:
   * Accede al archivo 016-chat con colores.html en tu navegador.

**3.3 Despliegue**

* **Local**:
  + Ejecutar el servidor como se describe arriba.
* **Producción**:
  + Configurar un servidor WSGI (como Gunicorn) para Flask.
  + Asegurar el acceso a 016-chat con colores.html.

**3.4 Pruebas y verificación**

* Probar las rutas API (/dame y /toma) usando herramientas como Postman.
* Verificar que los mensajes se actualicen en la interfaz.

**4. Mantenimiento y soporte**

**4.1 Manejo de errores**

* **Errores comunes**:
  + Código 500: Asegúrate de que las dependencias están instaladas.

**4.2 Registro de cambios (changelog)**

* **Versión 1.0**:
  + Implementación básica de chat.

**4.3 Soporte y contacto**

Para soporte técnico, escribe a: soporte@ejemplo.com.